

SMART AQUARIUM BERBASIS IOT



**Disusun sebagai salah satu syarat menyelesaikan Program Studi Strata I
pada Jurusan Informatika Fakultas Komunikasi dan Informatika**

Oleh :

ASMANDITYA HIBATULLAH

L200150012

**PROGRAM STUDI INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA**

2019

HALAMAN PERSETUJUAN

SMART AQUARIUM BERBASIS IOT

PUBLIKASI ILMIAH

oleh:

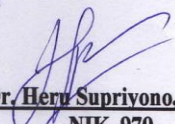
ASMANDITYA HIBATULLAH

L200150012

Telah diperiksa dan disetujui untuk diuji oleh:

Dosen Pembimbing

*See
pamun
31/07/2023*


Dr. Heru Supriyono, M.Sc.
NIK. 970

HALAMAN PENGESAHAN

SMART AQUARIUM BERBASIS IOT

OLEH

ASMANDITYA HIBATULLAH

1.200150012

Telah dipertahankan di depan Dewan Penguji

Fakultas Komunikasi dan Informatika

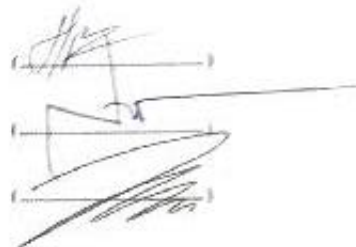
Universitas Muhammadiyah Surakarta

Pada hari Sabtu, 3 Juli 2019

dan dinyatakan telah memenuhi syarat

Dewan Penguji:

1. Heru Supriyono, S.T., M.Sc., Ph.D
(Ketua Dewan Penguji)
2. Nurgiyatna, S.T., M.Sc., Ph.D
(Anggota I Dewan Penguji)
3. Fajar Suryawan M.Eng. Sc, Ph.D
(Anggota II Dewan Penguji)



Dekan
Fakultas Komunikasi dan Informatika
Nurgiyatna, S.T., M.Sc., Ph.D
NIK. 881



Ketua Program Studi
Informatika
Dr. Heru Supriyono, M.Sc.
NIK. 970

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam naskah publikasi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

Surakarta, 31 Juli 2019

Penulis



ASMANDITYA HIBATULLAH

L200150012



**UNIVERSITAS MUHAMMADIYAH SURAKARTA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
PROGRAM STUDI INFORMATIKA**

Jl. A Yani Tromol Pos 1 Pabelan Kartasura Telp. (0271)717417, 719483 Fax (0271) 714448
Surakarta 57102 Indonesia. Web: <http://informatika.ums.ac.id>. Email: informatika@ums.ac.id

SURAT KETERANGAN LULUS PLAGIASI

No Surat 134/A.A-Il.3/INF-FK/VII/2019

Assalamu'alaikum Wr. Wb

Biro Skripsi Program Studi Informatika menerangkan bahwa :

Nama : Asmanditya Hibatullah
NIM : L200150012
Judul : Smart Aquarium Berbasis IoT
Program Studi : Informatika
Status : Lulus

Adalah benar-benar sudah lulus pengecekan plagiasi dari Naskah Publikasi Skripsi, dengan menggunakan aplikasi Turnitin.

Demikian surat keterangan ini dibuat agar dipergunakan sebagaimana mestinya.

Wassalamu'alaikum Wr. Wb

Surakarta, 23 juni 2019

Biro Skripsi Informatika

Ihsan Cahyo Utomo, S.Kom., M.Kom.



V

SMART AQUARIUM BERBASIS IOT

Abstrak

Kemajuan teknologi saat ini memudahkan untuk saling berkomunikasi, bertukar informasi antara sumber informasi dengan pengguna ataupun pencari informasi yang menggunakan fasilitas dari internet. Khususnya di bidang elektronik yang sekarang ini sudah mulai memasuki fase atau generasi dari IoT. Kekeruhan air memiliki dampak negatif bagi ikan dan tanaman dalam aquarium yakni ikan bisa mati karena keruhnya air dalam aquarium tersebut. Untuk itu dalam penulisan tugas akhir ini memiliki tujuan untuk memonitor kekeruhan air dalam aquarium dengan menggabungkannya dengan IoT, sebagai peringatan bahwa air dalam aquarium harus segera diganti. Metode yang digunakan dalam penulisan tugas akhir ini adalah eksperimental, dengan menguji sensor *turbidity* yang dimasukkan kedalam air jernih. *Hardware* yang digunakan adalah ESP32 sebagai mikrokontroller yang mengirim data sensor ke *database*, *turbidity* sensor sebagai sensor kekeruhan air, laptop Asus A455l dan software yang digunakan yakni Arduino IDE sebagai media pembuatan *source code* untuk diupload ke ESP32, Thingspeak.com sebagai platform untuk menyimpan data yang dikirimkan oleh ESP32. Setelah dilakukan pengujian pada sensor, langkah selanjutnya adalah pengujian *upload* data dari ESP32 ke Thingspeak.com. Dari percobaan *upload* data sebanyak 10 kali, tidak ada satupun yang gagal dalam *upload* data. Kesimpulannya adalah *upload* data ke Thingspeak.com memiliki presentase sebesar 100%, pengujian sensor dapat berjalan sesuai harapan, dan seluruh fungsi *hardware* dan *software* dapat berfungsi baik.

Kata Kunci: Aquarium, ESP32, IoT, mikrokontroller, Thingspeak.com, *turbidity* sensor

Abstract

Current technological advances facilitate communication with each other, exchange information between information sources with users or search for information using facilities from the internet. Especially in the electronics sector which has now started a phase or generation of IoT. I have negative water for fish and plants in the aquarium that is fish can die because of the water in the aquarium. To complete this final project, you must reactivate the aquarium water in the aquarium by supporting it with IoT, as a substitute it means that the water in the aquarium must be replaced immediately. The method used in completing this thesis is experimental, by asking for turbidity sensors that are put into clear water. The hardware used is ESP32 as a microcontroller that sends sensor data to the database, turbidity sensor as an air turbidity sensor, Asus A455l laptop and the software used is Arduino IDE as a media for making source code to be uploaded to ESP32, Thingspeak.com as a platform to store data transferred by ESP32. After testing the sensor, the next step is to upload data from ESP32 to Thingspeak.com. From the attempt to upload data 10 times, no attempt failed in uploading data. The

conclusion is uploading data to Thingspeak.com has a percentage of 100%, sensor testing can run as expected, and all hardware and software can be used properly.

Keywords: *Aquarium, ESP32, IoT, microcontroller, Thingspeak.com, turbidity sensor*

1. PENDAHULUAN

Kemajuan teknologi saat ini memudahkan untuk saling berkomunikasi, bertukar informasi antara sumber informasi dengan pengguna ataupun pencari informasi yang menggunakan fasilitas dari internet. Khususnya di bidang elektronik yang sekarang ini sudah mulai memasuki fase atau generasi dari IoT. Pada visi masa depan IoT akan menjadi utilitas dengan peningkatan kecanggihan dalam penginderaan, aktuasi, komunikasi, kontrol, dan menciptakan pengetahuan dari sejumlah besar data (Stankovic, 2014).

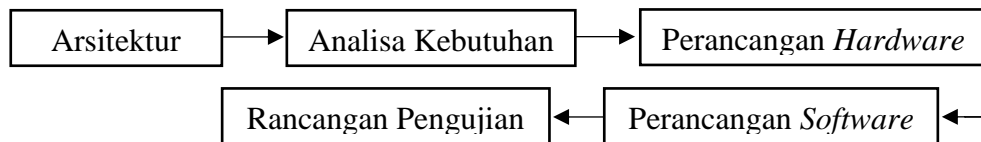
IoT merupakan sebuah paradigma bersifat inovatif yang berkembang secara pesat dalam pengaturan telekomunikasi nirkabel modern dengan cepat (Tahir, Kanwer, & Junaid, 2016). IoT diharapkan dapat menjadi sarana pengolahan data dari sensor atau peralatan elektronik yang terhubung dengan perangkat IoT secara *real time*. Dalam lingkungan IoT akan memungkinkan pengguna untuk mengelola dan mengoptimalkan peralatan elektronik dan peralatan listrik menggunakan internet. Diperkirakan dalam waktu dekat mayoritas komunikasi akan terjadi antara komputer dan peralatan elektronik lainnya yang akan terhubung satu sama lain dan akan bertukar informasi di antara mereka sehingga mengurangi interaksi manusia (Suresh, Daniel, Aswathy, & Parthasarathy, 2014).

Aquarium merupakan sebuah wadah untuk meletakkan ikan hias maupun tanaman air, akan tetapi seiring dengan berjalannya waktu aquarium tidak hanya diisi oleh ikan hias maupun tanaman air melainkan juga pasir, bebatuan agar terlihat lebih menarik pandangan mata. Dalam perawatannya ada beberapa kendala yang dihadapi oleh pemilik aquarium ini yakni kekeruhan air dalam aquarium. Kekeruhan dalam air memiliki dampak *negative* yang cukup besar terhadap ikan dan tanaman terhadap di aquarium.

Pesatnya perkembangan teknologi saat ini, kejernihan dalam air dalam aquarium dapat diketahui dengan bantuan dari sensor yang terhubung mikrokontroller sebagai pembaca data dari sensor. Dengan adanya sensor kejernihan air dapat diharapkan pengguna atau penggiat aquarium dapat terbantu dalam perawatan ikan hias dan tanaman air yang telah dipelihara.

2. METODE

Metodologi penelitian yang diterapkan dalam penyusunan tugas akhir *smart* aquarium berbasis IoT seperti pada Gambar 1.



Gambar 1. Tahapan Metodologi Penelitian

2.1 Arsitektur

Yang dimaksud dari *Smart* aquarium atau aquarium cerdas adalah aquarium yang dapat membantu meringankan perawatan yang dilakukan oleh pengguna. Dalam penelitian ini penulis membuat aquarium dengan spesifikasi sebagai berikut:

- Aquarium berukuran 31cm x 19cm x 25cm.
- Pompa filter air.
- Lampu LED berukuran 27cm x 6cm.

2.2 Analisa Kebutuhan

Penulis menganalisa kebutuhan berdasarkan apa yang dibutuhkan dalam pembuatan *smart* aquarium berbasis IoT untuk memonitor kejernihan air dalam aquarium menggunakan *turbidity* sensor

2.2.1 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan aplikasi ini:

- Arduino IDE versi 1.8.9, sebagai media penulisan kode program yang akan di *flash* ke mikrokontroller.
- <https://thingspeak.com/>, sebagai database untuk menyimpan data dari sensor.

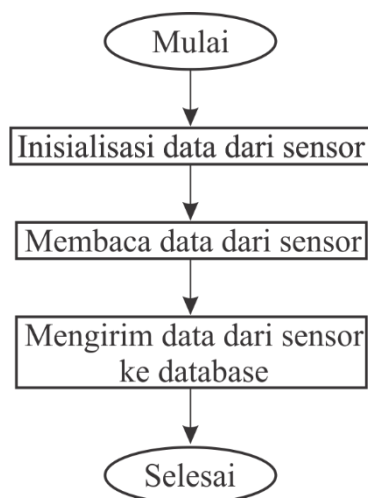
2.2.2 Analisis Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan dan uji coba aplikasi ini:

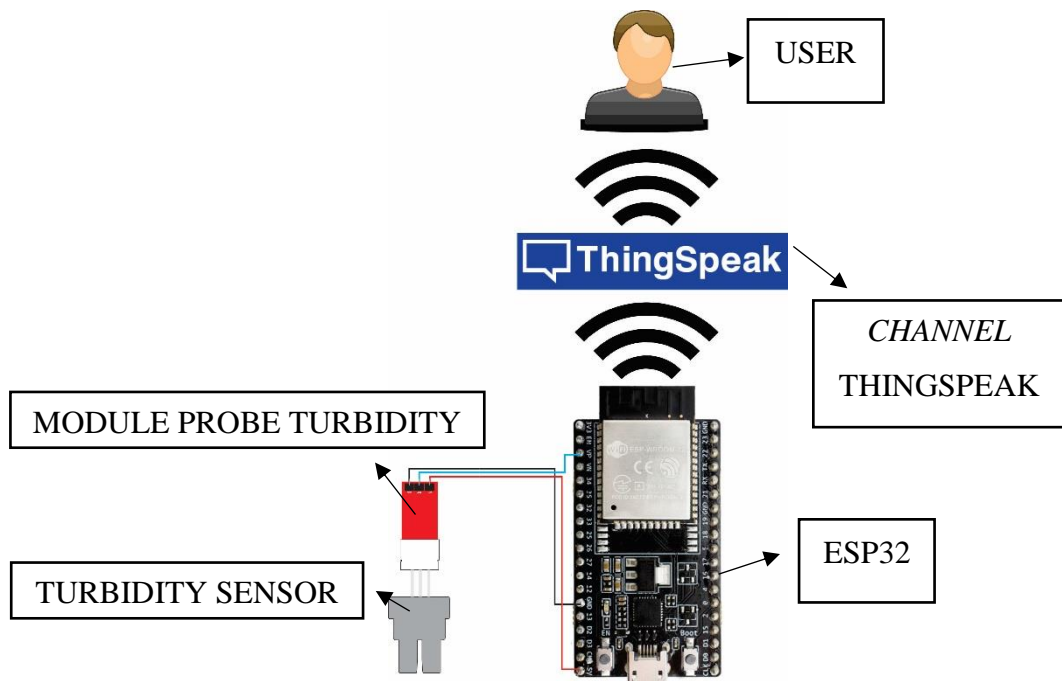
- 1) Laptop dengan spesifikasi *processor* 1.9 Ghz Intel Core i3 4030U, RAM 4Gb, SSD Sandisk Extreme II 120Gb SATA dan HDD 500Gb.
- 2) ESP32 sebagai mikrokontroller.
- 3) Sensor *Turbidity* sebagai sensor kejernihan air.
- 4) Kabel *jumper*.

2.3 Perancangan *Hardware*

Smart Aquarium berbasis IOT terdiri dari beberapa rangkaian *Hardware* yakni ESP32 sebagai mikrokontroller, dan *turbidity* sensor sebagai sensor kejernihan air. ESP32 sendiri merupakan serangkaian *chip* mikrokontroller berbiaya murah dan berdaya rendah yang memiliki kemampuan *Wi-Fi* dan *Bluetooth* (Singh & Kapoor, 2017). Pada *Turbidity Sensor* terdapat 3 pin yaitu 5V, *ground*, dan *out* yang akan dikoneksikan pada ESP32. *Out* disini merupakan pin analog yang dikoneksikan pada pin A0, untuk 5V dikoneksikan pada pin 5V ESP32, dan untuk pin *ground* dikoneksikan pada pin *GND* ESP32. Pada Gambar 2. *turbidity* sensor akan dimasukkan kepermukaan air aquarium untuk inialisasi data sensor kemudian ESP32 yang sudah terkoneksi dengan *Wi-Fi* akan membaca sinyal analog dari *turbidity* sensor, setelah memiliki data dari sensor maka data tersebut akan dikirimkan ke *channel* Thingspeak pengguna. Pada Gambar 3 dijelaskan bahwa ESP32 akan mengolah data dari *turbidity* sensor untuk kemudian mengirimkannya ke *channel* thingspeak pengguna. Setelah berhasil *upload* data ke channel, maka pengguna dapat melihat data secara langsung.

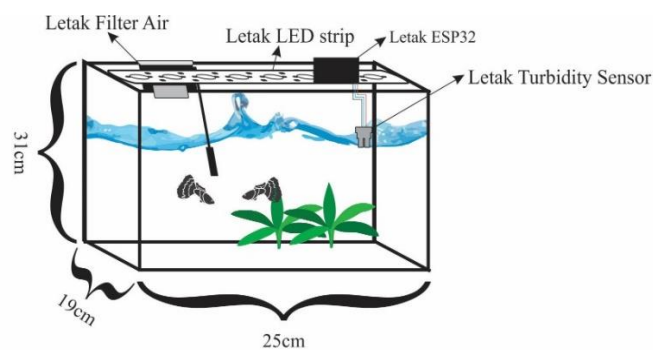


Gambar 2. *Flowchart Rancangan Hardware*



Gambar 3. Rancangan arsitektur *hardware*

Untuk rancangan implementasi *hardware*, letak ESP32 dan *module probe turbidity* akan dimasukkan kedalam satu box akrilik berukuran 11cm x 7cm x 7cm yang peletakannya akan ditempatkan di atas lampu LED berukuran 27cm x 6cm. Rancangan implementasi *hardware* dapat dilihat pada Gambar 4.



Gambar 4. Rancangan implementasi *hardware*

2.4 Perancangan *Software*

Pada tahap perancangan *software* terdiri dari beberapa tahapan, yakni pembuatan *source code* program untuk menjalankan fungsi sensor, menghubungkan

mikrokontroller dengan *Wi-Fi* agar dapat mengakses *Thingspeak.com* dan pembuatan *channels* pada *Thingspeak.com* sebagai penampil data dari sensor. *Thingspeak.com* dapat membuat aplikasi pencatatan sensor secara *real time* (Chwalisz, 2019). Sebelum pembuatan *channel*, pengguna harus terlebih dahulu membuat akun pada *Thingspeak.com* yang kemudian akan mendapatkan *API Keys* untuk dimasukkan kedalam *source code* mikrokontroller agar dapat melakukan pengiriman data dari sensor ke *Thingspeak.com*. Untuk rangkaian perancangan software dimulai dari inisialisasi data dari sensor kemudian mikrokontroller akan membaca data tersebut untuk dikirimkan ke *database* dari *channel Thingspeak.com* milik pengguna. Untuk dapat terkoneksi dengan *Wi-Fi*, dalam *source code* diberi perintah untuk melakukan koneksi ke *ssid Wi-Fi* terdekat dengan ESP32 dan memasukkan *password* dari *ssid* tersebut. Source code koneksi ESP32 ke *Wi-Fi* dapat dilihat pada Gambar 5.

```
Serial.begin(9600);
WiFi.disconnect();
WiFi.begin("Pakerte", "rasahdinggoyo");
while ((!(WiFi.status() == WL_CONNECTED))) {
    delay(300);
    Serial.print(".");
    pinMode(SENSOR, INPUT);
    Serial.println("Mulai");
}
```

Gambar 5. *Source code* koneksi *Wi-Fi*

Setelah ESP32 berhasil terkoneksi dengan *Wi-Fi* terdekat, langkah selanjutnya adalah membuka <https://thingspeak.com/> untuk pembuatan akun atau *Sign In* ke akun jika sudah mempunyai. Gambar 6. merupakan halaman awal dari <https://thingspeak.com/>. Pada Gambar 7. Merupakan tampilan *Sign In*.



Gambar 6. Halaman Awal *Thingspeak.com*

Gambar 7. Halaman *Sign In* *Thingspeak.com*

Setelah berhasil *Sign In* maka pengguna akan dialihkan ke menu *My Channels* yang merupakan halaman penampil pengkoleksian data dari perangkat *user*. Pada setiap *channel* memiliki API Key yang berbeda beda, karena API Key sendiri merupakan kode unik yang hanya dikeluarkan oleh suatu situs seperti Thingspeak untuk dapat diakses oleh ESP32. Halaman channel pengguna dapat dilihat pada Gambar 8.

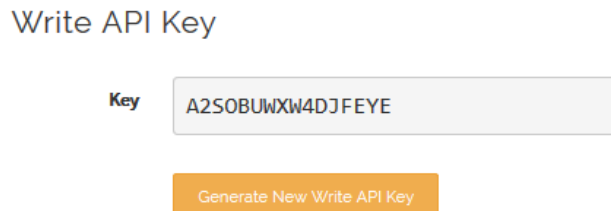
My Channels

[New Channel](#) [Q](#)

Name	Created	Updated
Turbidity Sensor Berbasis IOT	2019-06-24	2019-07-25 06:04
Private Public Settings Sharing API Keys Data Import / Export		

Gambar 8. Halaman *Channel* Pengguna Pada *Thingspeak.com*

Pada Gambar 9. Merupakan halaman yang menampilkan API Key dari *channel* pengguna. API Key pada Thingspeak dapat dibuat ulang dengan cara memilih opsi *Generate New Write API Key*.



Gambar 9. Halaman informasi API key

Setelah memiliki informasi API Key pada *channel*, langkah selanjutnya adalah memasukkan API Key pada *Source code* ESP32 untuk dapat mengirim data sensor ke *channel* pengguna. *Source code* Gambar 10. data sensor akan dikirimkan ke *channel* berdasarkan *field* yang berbeda beda. Field1 akan diisi dengan variabel *voltage*, field2 akan diisi dengan variabel *turbidity* dari hasil perhitungan pada sensor.

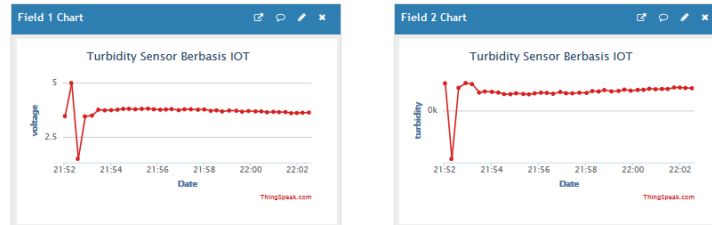
```
void kirim_thingspeak(float voltage, float turbidity) {
  if (client.connect("api.thingspeak.com", 80)) {
    request_string = "/update?";
    request_string += "key=";
    request_string += "A2SOBUWXW4DJFEYE";
    request_string += "&";
    request_string += "field1";
    request_string += "=";
    request_string += voltage;
    request_string += "&";
    request_string += "field2";
    request_string += "=";
    request_string += turbidity;
    Serial.println(String("GET ") + request_string + " HTTP/1.1\r\n" +
      "Host: " + thingSpeakAddress + "\r\n" +
      "Connection: close\r\n\r\n");
  }
}
```

Gambar 10. *Source code* mengirim data sensor ke Thingspeak.com

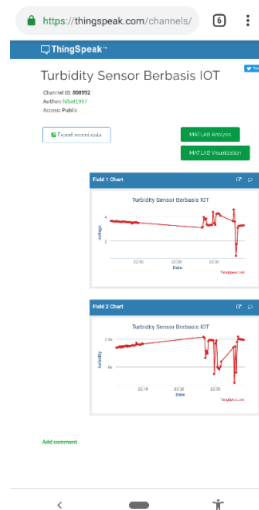
Jika berhasil upload data sensor ke *channel*, maka tampilan dari *channel stats* akan tampak seperti Gambar 11. Gambar 11. merupakan tampilan *channel stats* jika dilihat dari *web browser* pada laptop, sedangkan untuk tampilan *channel stats* pada perangkat android dapat diakses dengan membuka link <https://thingspeak.com/channels/808992>. 808992 merupakan *Channel ID* dari pengguna. Tampilan *channel stats* dapat dilihat pada Gambar 12.

Channel Stats

Created: [about a month ago](#)
Entries: 40



Gambar 11. Tampilan *Channel Stats* Pada Laptop Pengguna *Thingspeak.com*



Gambar 12. Tampilan *Channel Stats* Pada Android Pengguna *Thingspeak.com*

2.5 Pengujian Sistem

Rancangan pengujian kali ini didapatkan hasil dari pengujian 4 jenis air ditempatkan di wadah yang berbeda, sehingga *output* dari sensor juga berbeda beda.

3. HASIL DAN PEMBAHASAN

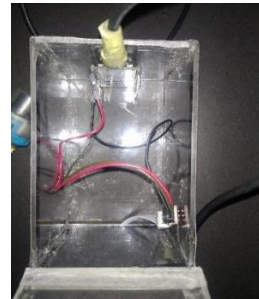
3.1 Rangkaian

Hasil dari penelitian ini adalah sebuah alat yang dapat monitor tingkat kekeruhan air dalam *aquarium* dengan menggunakan *Turbidity Sensor* dan *ESP32* yang sudah terkoneksi dengan *Thingspeak.com* sebagai database dari data sensor tersebut.

Tampak keseluruhan rangkaian pada Gambar 13a, dan untuk tampak dalam rangkaian dapat dilihat pada Gambar 13b.



(a)



(b)

Gambar 13. Rangkaian

3.2 Pengujian Sensor

Pengujian yang dilakukan oleh penulis adalah dengan melakukan percobaan ke air jernih sebanyak 14 kali. Dalam perhitungan satuan Nephelometric Turbidity Unit (NTU) memiliki rumus $y = (-26.7641)X + 135.0524$, dimana X merupakan hasil pembacaan nilai *voltage* pada sensor. Berdasarkan informasi pada WHO, air yang dikatakan jernih memiliki ambang batas kekeruhan adalah 0 NTU sampai dengan 5 NTU. Jika lebih dari 5 NTU maka air tersebut tidak bisa dikategorikan sebagai air jernih melainkan air cukup keruh. Hasil pengujian sensor dapat dilihat pada Tabel 1.

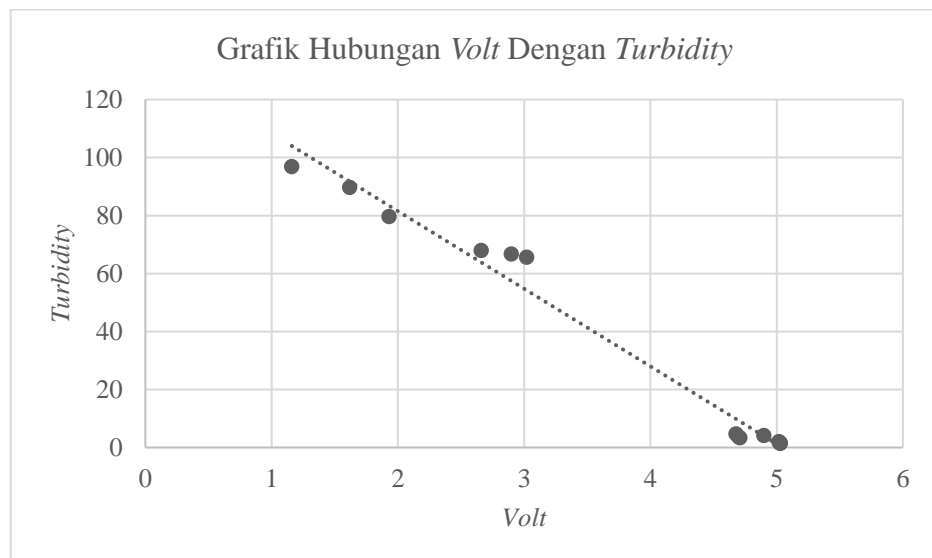
Tabel 1. Hasil Pengujian *Turbidity* Sensor

Percobaan Ke	Tegangan (<i>Volt</i>)	Kekeruhan (NTU)
1	4.9	4.1
2	5.02	2.05
3	5.03	1.51
4	5.03	1.51
5	5.03	1.51
6	4.68	4.71
7	4.71	3.31
8	4.7	3.66

Tabel 1. Hasil Pengujian *Turbidity* Sensor

9	3.02	65.6
10	2.9	66.7
11	2.66	67.88
12	1.93	79.57
13	1.62	89.63
14	1.16	96.82

Pada hasil pengujian sensor Tabel 1. dapat memperoleh grafik hubungan antara *volt* dengan kekeruhan dalam satuan NTU yang dapat dilihat pada Gambar 14.



Gambar 14. Grafik Hubungan *Volt* Dengan *Turbidity*

3.3 Pengujian *Upload* Data Thingspeak.com

Pengujian *upload* data dilakukan sebanyak 10 kali percobaan, dengan memperoleh hasil dapat diterima oleh Thingspeak.com dan memperoleh keberhasilan *upload* data sebesar 100%. Hasil pengujian *upload* data dapat dilihat pada tabel 2.

Tabel 2. *Upload Data*

Dibuat Pada	No	field1(V)	field2(Turbidity)
2019-07-30 19:18:39 UTC	1	5.03	1.51
2019-07-30 19:18:44 UTC	2	5.03	1.51
2019-07-30 19:18:49 UTC	3	5.03	1.51
2019-07-30 19:18:54 UTC	4	5.02	2.05
2019-07-30 19:18:59 UTC	5	4.9	4.1
2019-07-30 19:19:04 UTC	6	4.68	4.71
2019-07-30 19:19:09 UTC	7	4.71	3.31
2019-07-30 19:19:14 UTC	8	4.7	3.66
2019-07-30 19:19:19 UTC	9	3.02	65.6
2019-07-30 19:19:24 UTC	10	2.9	66.7

4. PENUTUP

Berdasarkan hasil pengiriman data dari sensor ke *Thingspeak.com*, maka disimpulkan bahwa pengguna dapat memonitor seberapa keruh air pada aquarium secara langsung tanpa harus berada di dekat aquarium pengguna dan presentase keberhasilan *upload* data sebesar 100%. Kesimpulan penelitian ini adalah seluruh perangkat *hardware* dan *software* dapat berfungsi sesuai yang diharapkan oleh penulis.

Smart aquarium berbasis *IOT* memiliki kekurangan yakni penggunaan sumber daya listrik, jika listrik yang dipakai oleh pengguna padam maka fungsi dari *Smart Aquarium berbasis IOT* ini akan terhenti. Maka dari itu solusi dari kekurangan ini adalah dengan menambahkan baterai bertegangan 5V sebagai sumber daya jika listrik padam. Pada *turbidity* sensor harus sering untuk dikalibrasi, sehingga pembacaan sensor dapat memperoleh galat yang sedikit.

DAFTAR PUSTAKA

- Chwalisz, M. (2019). *ThingSpeak Documentation*. Retrieved from <https://buildmedia.readthedocs.org/media/pdf/thingspeak/latest/thingspeak.pdf>
- Singh, K. J., & Kapoor, D. S. (2017). Create Your Own Internet of Things: A survey of IoT platforms. *IEEE Consumer Electronics Magazine*, 6(2), 57–68.

<https://doi.org/10.1109/MCE.2016.2640718>

Stankovic, J. A. (2014). Research directions for the internet of things. *IEEE Internet of Things Journal*, 1(1), 3–9. Retrieved from <http://www.intercom.virginia.edu/~stankovic/psfiles/IOT.pdf>

Suresh, P., Daniel, J. V., Aswathy, R. H., & Parthasarathy, V. (2014). A state of the art review on the Internet of Things (IoT). *International Conference on Science, Engineering and Management Research (ICSMEMR 2014)*. Retrieved from https://www.researchgate.net/publication/275529109_A_State_of_the_art_review_on_the_Internet_of_Things_IoT_History_Technology_and_fields_of_deployment

Tahir, H., Kanwer, A., & Junaid, M. (2016). Internet of Things (IoT): An Overview of Applications and Security Issues Regarding Implementation. *International Journal of Multidisciplinary Sciences and Engineering*, 7(1), 14–22. Retrieved from <http://www.ijmse.org/Volume7/Issue1/paper3.pdf>